- Converting a Sum Term to Standard POS

Each sum term in a POS expression that does not contain all the variables in the domain can be expanded to standard form to include all variables in the domain and their complements. As stated in the following steps, a nonstandard POS expression is converted into standard form using Boolean algebra **Rule8** $(A.\overline{A}=0)$ **:A variable multiplied by its complement equal 0.**

Step1: Add to each nonstandard product term a term made up of the product of ,the missing variable and its complement. This results in two sum terms .As you know ,you can add 0 to anything without changing its value.

Step2: Apply **rule12 : A +BC =(A+B)(A+C)**

Step3: Repeat Step 1 until al resulting sum terms contain all variables in the domain in either complement or un complemented form.

Example **: Convert the following Boolean expression into Standard**

POS form. $(A+\overline{B}+C)(\overline{B}+C+\overline{D})(A+\overline{B}+\overline{C}+D)$

Solution: the domain of this POS expression is A,B,C,D. take one term at a time,

The first term $A+\overline{B}+C$ *D or* $\overline{D}$ *, so add* $D\overline{D}$ *and apply **rule12** as follows:*

$$A+\overline{B}+C=(A+\overline{B}+C)+D\overline{D}=(A+\overline{B}+C+D).(A+\overline{B}+C+\overline{D})$$

The Second term $\overline{B}+C+\overline{D}$ *is missing variable A or* $\overline{A}$ *,so add* $A\overline{A}$ *and apply rule 12 as follows:*

$$\overline{B}+C+\overline{D}=(\overline{B}+C+\overline{D})+A\overline{A}=(\overline{B}+C+\overline{D}+A).(\overline{B}+C+\overline{D}+\overline{A})$$

***The third term ,** $A+\overline{B}+\overline{C}+D$, is already in standard form , so, the standard POS form of the original expression is as follows:*

$(A+\overline{B}+C)(\overline{B}+C+\overline{D})(A+\overline{B}+\overline{C}+D)=$
$(A+\overline{B}+C+D).(A+\overline{B}+C+\overline{D}).(\overline{B}+C+\overline{D}+A).(\overline{B}+C+\overline{D}+\overline{A}).$
$A+\overline{B}+\overline{C}+D$

✍ **Home work**

Convert the expression $(A+\overline{B})(B+C)$ to Standard POS Form

- Binary Representation of a Standard Sum Term

A standard sum term is equal to 0 for only one combination of variable values. For example ,the sum term $A+\overline{B}+C+\overline{D}$ is 0 when A=0 , B=1 , C=0 , and D=1 , as shown below , and 1 for all for all other combinations of values for the variables ,

$$A+\overline{B}+C+\overline{D}=0+\overline{1}+0+\overline{1}=0+0+0+0=0$$

In this case ,the sum term has a binary value of 0101 (decimal 5) .

☞**Remember** , a sum term is implemented with an OR gate whose output is 0 only if each its inputs is 0. Inverters are used to produce the complements of the variables as required .

A POS expression is equal to 0 only if one or more of the sum terms in the expression is equal to 0.

Example: Determine the binary values of the variables for which the following standard POS expression is equal to 0:

$$(A+B+C+D)(A+\overline{B}+\overline{C}+D)(\overline{A}+\overline{B}+\overline{C}+\overline{D})$$

Solution:

The first term $(A+B+C+D)$ *is equal to 0 when A=0 , B=0 , C=0 and D=0.*

$$A+B+C+D=0+0+0+0=0$$

The second term $A+\overline{B}+\overline{C}+D$ *is equal to 0 when A=0 , B=1, C=1 and D=1*

$$A+\overline{B}+\overline{C}+D=0+\overline{1}+\overline{1}+0=0+0+0+0=0$$

The third term $\overline{A}+\overline{B}+\overline{C}+\overline{D}$ *is equal to 0 when A=1 ,B=1 , C=1 and D=1*

$$\overline{A}+\overline{B}+\overline{C}+\overline{D}=\overline{1}+\overline{1}+\overline{1}+\overline{1}=0+0+0+0=0$$

The POS expression equals 0 when any of the three terms eaual 0.

✍ Home work

Determine the binary values for which the following POS expression is equal to 0 : $(X+\overline{Y}+Z)(\overline{X}+Y+Z)(X+Y+\overline{Z})(\overline{X}+\overline{Y}+\overline{Z})(X+\overline{Y}+\overline{Z})$ **is this a   Standard POS expression?**

♦ Convert SOP to POS form

The binary values of the product term in a given SOP expression are not present in the equivalent standard POS expression .Also , the binary values that are not represented in the SOP expression are present in the equivalent POS expression. Therefore ,to convert from standard SOP to standard POS, the following steps are taken:

Step1: Evaluate each product term in the SOP expression . That is , determine the binary numbers that represent the product term.

Step2: Determine all of the binary numbers not included in the evaluation in step1.

Step3: Write the equivalent sum term for each binary number from step2 and express in POS form.

Note: using a similar procedure , you can go from POS to SOP .

Example : Convert the following SOP expression to an equivalent POS expression:

$$\overline{A}\overline{B}\overline{C} + \overline{A}B\overline{C} + \overline{A}BC + A\overline{B}C + ABC$$

Solution: *the evaluation is as follows:* **000+010+011+101+111**

Since there are three variables in the domain of this expression, there afe a total of eight(2^3) possible combinations.

The SOP expression contains five of these combination , so the POS must contain the other three which are 001 ,100 and 110.

Remember , these are the binary values that take the sum term 0. the equivalent POS expression is $(A + B + \overline{C})(\overline{A} + B + C)(\overline{A} + \overline{B} + C)$

🌢 Boolean Expression And Truth Tables

- Converting SOP expressions to Troth Table Format

Recall from pervious sections , that an SOP expression is equal to 1 only if at least one of the product terms is equal to 1.

- The first step un constructing a truth table is to list all possible combinations of binary values of the variables in the expression.

- Next ,convert the SOP expression to standard form if it is not already.

- Finally ,place a 1 in the output column (X) for each binary value that takes the standard SOP expression a 1 and place 0 for all remaining binary values.

This procedure illustrated in the example bellow :

Example :Develop a truth table for the Standard SOP expression

$$\overline{A}\,\overline{B}C + A\overline{B}\,\overline{C} + ABC$$

Solution

There are three variables in the domain, so there are **eight** possible combinations of binary values of the variables as listed in the left three columns of tables bellow .

The binary values that make the product terms in the expressions equal to 1 are $\overline{A}\,\overline{B}C : 001 ; A\overline{B}\,\overline{C} : 100 ; and\ ABC : 111$. For each of these binary values , place a 1 in the output column as shown in the table. For each of the remaining binary combinations , place a 0 in the output column.

Input			Output	Product term
A	B	C	X	
0	0	0	0	
0	0	1	1	$\overline{A}\,\overline{B}C$
0	1	0	0	
0	1	1	0	
1	0	0	1	$A\overline{B}\,\overline{C}$
1	0	1	0	
1	1	0	0	
1	1	1	1	ABC

- Converting SOP expressions to Troth Table Format

Recall from previous sections, that a POS expression is equal to 0 only if at least one of the sum terms is equal to 0.

- To construct a truth table from a POS expression ,list all the possible combinations of binary values of the variable just as done for SOP expression.

- Next, convert the POS expression to standard form if it is not already.

- Finally, place a 0 in the output column (X) for each binary value that makes the expression a 0 and place a 1 for all the remaining binary values.

This procedure is illustrated in example bellow:

Example: determine the troth table for the following standard POS   expression:

$$(A+B+C)(A+\overline{B}+C)(A+\overline{B}+\overline{C})(\overline{A}+B+\overline{C})(\overline{A}+\overline{B}+C)$$

Solution

There are three variables in the domain, so there are **eight** possible combinations of binary values of the variables as listed in the left three columns of tables bellow .

The binary values that make the product terms in the expressions equal to 0 are

$$A+B+C:000 \;\; ; \;\; A+\overline{B}+C:010 \;\; ; \;\; A+\overline{B}+\overline{C}:011;$$
$$\overline{A}+B+\overline{C}:101; \;\; and \;\; \overline{A}+\overline{B}+C:100$$

For each of these binary values , place a 0 in the output column as shown in the table. For each of the remaining binary combinations , place a 1 in the output column.

Input			Output	Sum term
A	B	C	X	
0	0	0	0	$A+B+C$
0	0	1	1	
0	1	0	0	$A+\overline{B}+C$
0	1	1	0	$A+\overline{B}+\overline{C}$
1	0	0	1	

1	0	1	0	$\overline{A}+B+\overline{C}$
1	1	0	0	$\overline{A}+\overline{B}+C$
1	1	1	1	

✐ Home work

1- **Create a truth table for the standard SOP expression :** $\overline{A}B\overline{C}+A\overline{B}\overline{C}$

2- **Create a troth table for the standard POS expression :**

$$(A+\overline{B}+C)(A+B+\overline{C})(\overline{A}+\overline{B}+\overline{C})$$

- ## Determining Standard Expressions from Troth Table

To determine the standard SOP represented by a truth table.

- List the binary value of the input variables for which the outputs is 1.

- Convert each binary value to the corresponding product term by replacing each 1 with the corresponding variable and each 0 with the corresponding variable complement.

For example, the binary value **1010** is converted to a product term as follows:

$$1010 \ \rightarrow A\overline{B}C\overline{D}$$

If you substitute , you can see that the product term is 1:

$$A\overline{B}C\overline{D}=1.\overline{0}.1.\overline{0}=1.1.1.1=1$$

To determine the standard POS represented by a truth table.

- List the binary value of the input variables for which the outputs is 0.

- Convert each binary value to the corresponding sum term by replacing each 1 with the corresponding variable complement and each 0 with the corresponding variable .

For example, the binary value **1010** is converted to a sum term as follows:

$$1001 \rightarrow \overline{A} + B + C + \overline{D}$$

If you substitute , you can see that the sum term is 0:

$$\overline{A} + B + C + \overline{D} = \overline{1} + 0 + 0 + \overline{1} = 0 + 0 + 0 + 0 = 0$$

🌢 The KARNUPH Map

- Introduction

A karnauph map provides a systematic method for simplifying Boolean expressions and, if properly used ,will produce the simplest SOP or POS expression possible, known as *the minimum expression*. As you have seen , the effectiveness of the algebraic simplification depends on your familiarity with the laws, rules , and theorems of Boolean algebra and the on your ability to apply them. The Karnauph map on other hand , provide a "cookbook" method for simplification . other simplification techniques include the *Quine-McClusky* method and the **Espresso algorithm**

-A Karnauph map

Is similar to a truth table because it present all of the possible value of input variables and the resulting output for each value.

- ❖ Instead of being organized into columns and rows like a troth table, the Karnauph map is an array of **cells** in which each cell represents a binary value of the input variables.

❖ The cells are arranged in a way so that simplification of a given expression is simply a matter of properly grouping the cells.

❖ Karnauph maps can be used for expressions with two , three , for , and five variables, but we will discuss only 2-variable , 3-variables and 4-variables situations to illustrate the principles.

❖ The number of cells in Karnauph map, as well as the number of rows in a troth table, is equal to the total number of possible input variable combinations.

❖ For two variable , the number of cells is 2^2=4 cells , for three variable 2^3=8 cells , and four variables 2^4=16 cells .

-A 2- Variable Karnauph map

The 2-variable karnauph map is an array of four cells, as shown in table(1) bellow .

▪ In this case A and B are used for the variables although other letters could be used.

▪ Binary values of A are along the left side and the values of B are across the top.

▪ The value of a given cell is the binary values of A at the left in the same row combined with the value of B at the to in the same column.

For example, the cell in the upper left corner has a binary value 00 and the cell in the lower right corner has a binary value 11 . Table(2) shows the standard product terms that are represented by each cell in the 2-varable Karnaugh map.

	Table(1)			**Table(2)**

A \ B	**0**	**1**
0	00	01
1	10	11

A \ B	**0**	**1**
0	$\overline{A}\,\overline{B}$	$\overline{A}B$
1	$A\overline{B}$	AB

-A 3- Variable Karnauph map

The 3-variable karnauph map is an array of eight cells, as shown in table(1) bellow .

- In this case A ,B and C are used for the variables although other letters could be used.

- Binary values of A and B are along the left side and the values of C are across the top.

- The value of a given cell is the binary values of A and B at the left in the same row combined with the value of C at the to in the same column.

For example, the cell in the upper left corner has a binary value 000 and the cell in the lower right corner has a binary value 101 . Table(2) shows the standard product terms that are represented by each cell in the 3-varable Karnaugh map.

	Table(1)	
C **AB**	**0**	**1**
00	000	001
01	010	011
11	110	111
10	100	101

	Table(2)	
C **AB**	**0**	**1**
00	$\overline{A}\,\overline{B}\,\overline{C}$	$\overline{A}\,\overline{B}C$
01	$\overline{A}B\overline{C}$	$\overline{A}BC$
11	$AB\overline{C}$	ABC
10	$A\overline{B}\,\overline{C}$	$A\overline{B}C$

- A 4- Variable Karnauph map

The 4-variable karnauph map is an array of eight cells, as shown in table(1) bellow .

▪ In this case A ,B C ,and D are used for the variables although other letters could be used.

▪ Binary values of A and B and C and D are along the left side and the values of C are across the top.

- The value of a given cell is the binary values of A and B at the left in the same row combined with the value of C and D at the to in the same column.

For example, the cell in the upper right corner has a binary value 0010 and the cell in the lower right corner has a binary value 1010 . Table(2) shows the standard product terms that are represented by each cell in the 4-varablr Karnaugh map.

Table(1)

AB \ CD	00	01	11	10
00	0000	0001	0011	0010
01	0100	0101	0111	0110
11	1100	1101	1111	1110
10	1000	1001	1011	1010

Table(2)

AB \ CD	00	01	11	10
00	$\overline{A}\,\overline{B}\,\overline{C}\,\overline{D}$	$\overline{A}\,\overline{B}\,\overline{C}D$	$\overline{A}\,\overline{B}CD$	$\overline{A}\,\overline{B}C\overline{D}$
01	$\overline{A}B\overline{C}\,\overline{D}$	$\overline{A}B\overline{C}D$	$\overline{A}BCD$	$\overline{A}BC\overline{D}$
11	$AB\overline{C}\,\overline{D}$	$AB\overline{C}D$	$ABCD$	$ABC\overline{D}$
10	$\overline{C}B\overline{C}\,\overline{D}$	$\overline{C}B\overline{C}D$	$A\overline{B}CD$	$A\overline{B}C\overline{D}$

- **Cell Adjacency**

The cells in a Karnauph map are arranged so that there is only a single-variable between adjacent cells. **Adjacency is defined by a single-variable change.**

In The 2-Variable map , the 00 cell is adjacent to the 10 cell, and the 10 cell but not adjacent to 11 cell.

In the 3-Varibale map, the 010 cell is adjacent to 000 cell, and the 110 cell. the 010 cell is not adjacent to the 001 cell, the 100, or the 101 cell.

Note1: Physically ,each cell is adjacent to the cells that are immediately next to it on any of its four sides.

Note2: A cell is not adjacent to the cells that diagonally touch any of its corners.

Notes3: Also, the cells in the top row are adjacent to the corresponding cells in the bottom row and the cells in the outer left column are adjacent to the corresponding cells in the outer right column. *This is called "wrap-around" adjacency because you can think of the map as wrapping around from top to bottom to form a cylinder or from left to right to form a cylinder.*

Correct adjacency   **wrong adjacency**
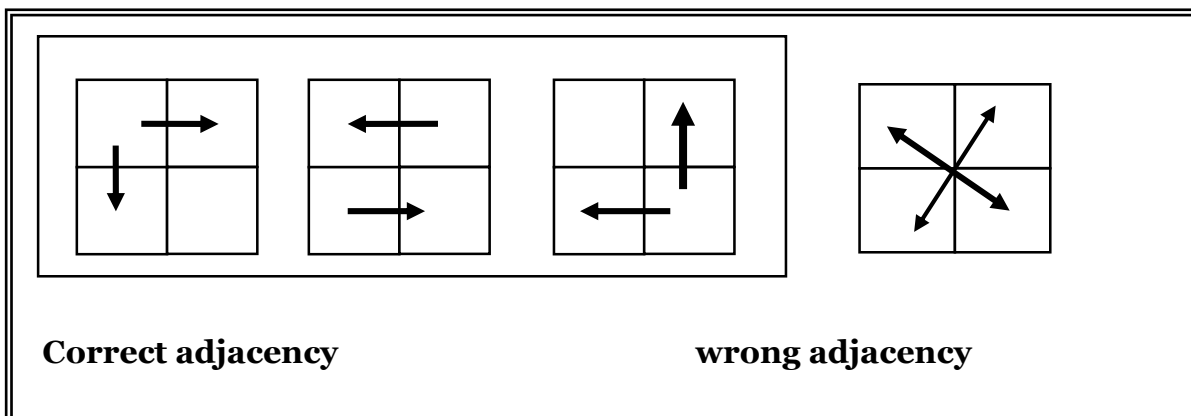
2-variable adjacency
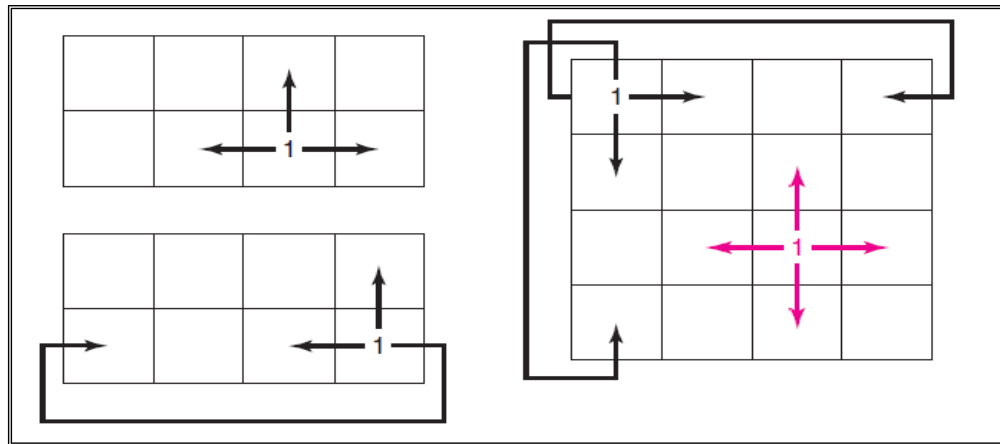
3 and 4 –variable adjacency

We will explain the way to use karnauph map to minimize SOP standard expression only .

⬦ KARNAUGH Map SOP Minimization

As stated in the last section, the Karnaugh map is used for simplifying Boolean expressions to their minimum form. A minimized SOP expression contains the fewest possible terms with the fewest possible variables per term.

Generally, a minimum SOP expression can be implemented with fewer logic gates than a standard expression .

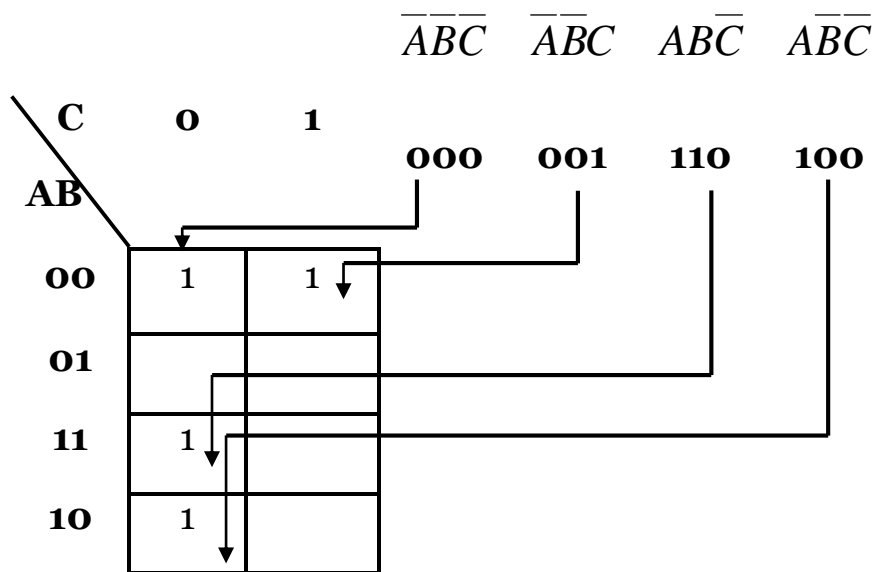## - Mapping a standard SOP Expression

- For an SOP expression in standard form, a 1 is placed on the Karnaugh map for each product term in the expression.

- Each 1 is placed in a cell corresponding to the value of a product term. For example, for the product term $A\bar{B}C$,a 1 goes in the 101 cell on a 3-variablemap.

- When an SOP expression is completely mapped, there will be a number of 1's on the Karnaugh map equal to the number of product terms in the standard SOP expression.

- The cells that do not have a 1 are the cells for which the expression is 0.

- Usually ,when working with SOP expressions, the 0's are left off the map.

The following steps and the illustration in the next figure show the mapping process .

Step1: Determine the binary value of each product term in the standard SOP expression. After some practice ,you can usually do the evaluation of terms mentally.

Step2: As each product term is evaluated, place a 1 on the Karnaugh map in the cell having the same value as the product term.

$$\overline{A}\,\overline{B}\,\overline{C} \quad \overline{A}\,\overline{B}\,C \quad AB\overline{C} \quad A\overline{B}\,\overline{C}$$

C \\ AB	0	1				
			000	001	110	100
00	1	1				
01						
11	1					
10	1					

✐ **Home work**

Map the following standard SOP expression on a Karnauph map

$$1 - \overline{A}\overline{B}C + \overline{A}B\overline{C} + AB\overline{C} + ABC$$

$$2 - \overline{A}BC + A\overline{B}C + A\overline{B}\overline{C}$$

$$3 - \overline{A}\overline{B}CD + \overline{A}B\overline{C}\overline{D} + AB\overline{C}D + ABCD + AB\overline{C}\overline{D} + \overline{A}\overline{B}CD + A\overline{B}C\overline{D}$$